



PATENT ABSTRACTS OF JAPAN

(11) Publication number: **09074359 A**(43) Date of publication of application: **18 . 03 . 97**

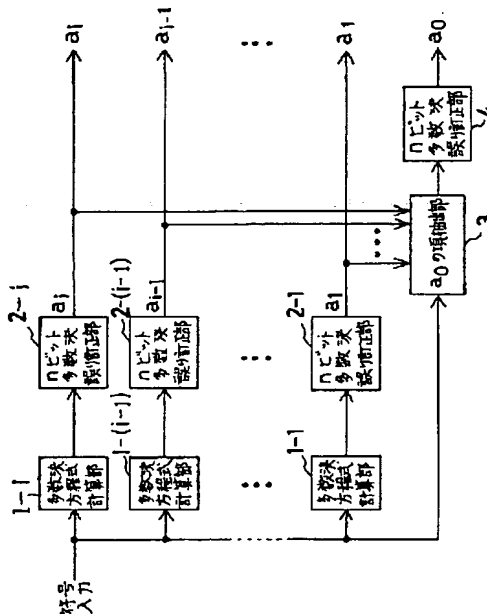
(51) Int. Cl.

H03M 13/00
G06F 11/10(21) Application number: **07226414**(22) Date of filing: **04 . 09 . 95**(71) Applicant: **FUJITSU LTD**(72) Inventor:
TAJIMA KAZUYUKI
KAWAI MASAOKI
SHINOMIYA TOMOHIRO
ABIRU SETSUO
HIROTA MASAKI
MIYABE MASATAKE(54) **ERROR CORRECTION AND DECODING CIRCUIT** COPYRIGHT: (C)1997,JPO

(57) Abstract:

PROBLEM TO BE SOLVED: To enable a high-speed operation without using any ROM while using a simple logic circuit for decoding a Read-Muller code.

SOLUTION: Majority equation calculation parts from 1-1 to 1-i respectively composed of $2S^{-1}$ two-input adder circuits provided corresponding to respective terms of element information, a_0 to a_i [a_0 is least significant] are provided for parallelly inputting respective bit signals ($x_0 \dots x_p$) of an input signal (x) so as to perform error correction and decoding by receiving the signal (x) for which the code length is $2S$ and the respective terms a_0 to a_i of element information are made into binary linear Read-Muller codes according to the expression $x = a_0v_0 + \dots + a_1v_1 + a_{i-1}v_{i-1}$ with v_0 to v_i as respective linear bases. For $2S^{-1}$ pieces of respective outputs from those parts, the error correction of n ($=2S^{-2}-1$) bits is performed at respective n -bit majority error correction parts from 2-1 to 2-i, codes a_1 to a_i are generated, the term of a_0 is extracted at a term extraction part 3 for a_0 by the input signal and the respective codes a_1 to a_i and next, error correction is performed by an n -bit majority error correction part 4.



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-74359

(43) 公開日 平成9年(1997)3月18日

| | | | | |
|---------------------------|-------|--------|---------------|---------|
| (51) Int.Cl. ⁸ | 識別記号 | 庁内整理番号 | F I | 技術表示箇所 |
| H 0 3 M 13/00 | | | H 0 3 M 13/00 | |
| G 0 6 F 11/10 | 3 3 0 | | G 0 6 F 11/10 | 3 3 0 S |

審査請求 未請求 請求項の数 5 O L (全 23 頁)

(21) 出願番号 特願平7-226414

(22) 出願日 平成7年(1995)9月4日

(71) 出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中4丁目1番1号

(72) 発明者 田島 一幸

神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(72) 発明者 河合 正昭

神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(74) 代理人 弁理士 穂坂 和雄 (外2名)

最終頁に続く

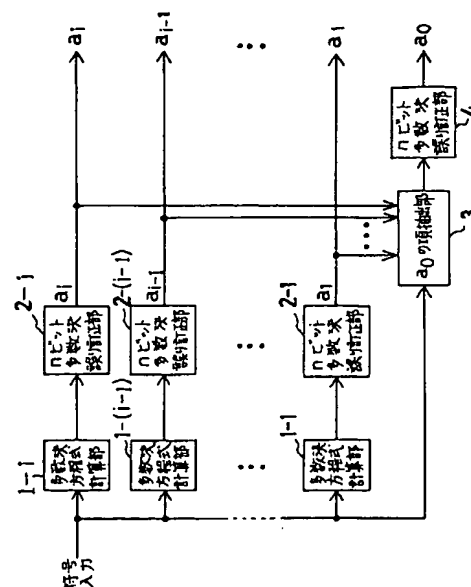
(54) 【発明の名称】 誤り訂正復号回路

(57) 【要約】

【課題】本発明は誤り訂正復号回路に関し、リード・マラー符号の復号を簡単な論理回路を用いてROMを使用せずに高速動作を可能にすることを目的とする。

【解決手段】符号長が 2^s で元の情報の各項 $a_0 \sim a_1$ (a_0 が最下位)が $v_0 \sim v_1$ を一次の各基底として式 $x = a_0 v_0 + a_1 v_1 + \dots + a_{s-1} v_{s-1}$ により2元一次のリード・マラー符号化された信号 x を受け取って誤り訂正と復号を行うため、入力信号 x の各ビット信号($x_0 \dots x_{s-1}$)が並列に入力され、 $a_1 \sim a_{s-1}$ の各項に対応して設けたそれぞれ2入力の 2^{s-1} 個の加算回路で構成された多数決方程式計算部を備える。そこからの 2^{s-1} 個の各出力は各 n ビット多数決誤り訂正部で n ($= 2^{s-1} - 1$)ビットの誤り訂正を行って符号 $a_1 \sim a_{s-1}$ を発生し、 a_0 の項抽出部で入力信号と各符号 $a_1 \sim a_{s-1}$ により a_0 項を抽出し、次に n ビット多数決誤り訂正部で誤り訂正を行うよう構成する。

本発明の原理構成図



【特許請求の範囲】

【請求項1】 符号長が 2^s で元の情報の各項 $a_0 \sim a_{p-1}$ (a_0 が最下位)が $v_0 \sim v_{p-1}$ を零次および一次の各基底として次の式

$$x = a_0 \cdot v_0 + a_1 \cdot v_1 + \dots + a_{p-1} \cdot v_{p-1} + a_p \cdot v_p$$

により2元一次のリード・マラー符号化された信号 x を受け取って誤り訂正と復号を行う誤り訂正復号回路において、入力信号 x の各ビット信号(x_0, \dots, x_{p-1} ; $p = 2^s - 1$)が並列に入力され、 $a_1 \sim a_{p-1}$ の各項に対応して設けられそれぞれ2入力の 2^{s-1} 個の加算回路で構成された多数決方程式計算部と、前記各多数決方程式計算部からの 2^{s-1} 個の出力が入力されてそれぞれ n ビット($n = 2^{s-2} - 1$)の多数決誤り訂正出力を発生してそれぞれ符号 $a_1 \sim a_{p-1}$ の各出力を発生する n ビット多数決誤り訂正部と、入力信号と前記各 n ビット多数決誤り訂正部の全ての出力とが入力されて符号 a_0 項を抽出する a_0 の項抽出部と、その出力である 2^s 個の信号から誤り訂正した a_0 を発生する n ビット多数決誤り訂正部とで構成されることを特徴とする誤り訂正復号回路。

【請求項2】 符号長が 2^s で元の情報の各項 $a_0 \sim a_{p-1}$ (a_0 が最下位)の符号が $v_0 \sim v_{p-1}$ を一次の各基底として次の式

$$x = a_0 \cdot v_0 + (a_1 + a_0) \cdot v_1 + (a_2 + a_0) \cdot v_2 + \dots + (a_{p-1} + a_0) \cdot v_{p-1} + (a_p + a_0) \cdot v_p$$

により2元一次のリード・マラー符号化された信号 x を受け取って誤り訂正と復号を行う誤り訂正復号回路において、請求項1に記載の各項 $a_1 \sim a_{p-1}$ に対応する前記多数決方程式計算部と、前記多数決方程式計算部に対応して設けられた各 n ビット多数決方程式計算部と、前記 a_0 の項抽出部とその出力が入力されて誤り訂正された a_0 を発生する n ビット多数決誤り訂正部とを備え、前記 $a_1 \sim a_{p-1}$ に対応する各 n ビット多数決方程式計算部からの各出力と、前記 a_0 を発生する n ビット多数決誤り訂正部の出力とが入力される各排他的論理と手段を設けたことを特徴とする誤り訂正復号回路。

【請求項3】 請求項1または2において、前記複数の多数決方程式計算部の出力を時分割で選択するセレクタと、前記セレクタの出力を受け取って n ビット多数決の誤り訂正を行い各項の出力を順番に発生する1個の n ビット多数決誤り訂正部を設けたことを特徴とする誤り訂正復号回路。

【請求項4】 請求項1乃至3に記載の誤り訂正回路の入力側に、上記符号長が 2^s のリード・マラー符号の上位ビットをラッチして前記誤り訂正回路へ出力する上位ラッチと、下位ビットをラッチする下位ラッチと、前記下位ラッチの出力と、符号長が 2^{s-1} で1ビットを表し、各要素が互いに排他的な関係をもつ 2^{s-1} 多数決符号の情報“0”を表す符号とが入力される下位セレクタとを備え、符号長が 2^s の2元一次のリード・マラー符

号化された信号を復号する時、前記下位セレクタで下位ラッチの出力を選択して、前記上位ラインの出力と共に前記誤り訂正回路へ入力して復号と誤り訂正を行い、符号長が 2^s の上位に前記 2^{s-1} 多数決符号が受信されると、前記下位セレクタを前記 2^{s-1} 多数決符号の情報“0”を選択するよう切り替えて、前記誤り訂正回路の中の上位ビットと下位ビットの排他的論理和を行う多数決方程式計算部の出力から前記 2^{s-1} 多数決符号の誤り訂正出力を得ることを特徴とする誤り訂正復号回路。

10 【請求項5】 請求項4において、前記上位ラッチの出力と符号長が 2^{s-1} で1ビットを表し、各要素が互いに排他的な関係をもつ 2^{s-1} 多数決符号の情報“0”を表す符号とが入力されるセレクタと、下位ビットをラッチして誤り訂正回路へ出力する下位ラッチとを備え、符号長が 2^s の下位に前記 2^{s-1} 多数決符号が受信されると、前記上位セレクタを前記 2^{s-1} 多数決符号の情報“0”を選択するよう切り替えることを特徴とする誤り訂正復号回路。

【発明の詳細な説明】

20 【0001】

【発明の属する技術分野】本発明は多ビット誤りを訂正できるリード・マラー符号等の誤り訂正復号回路に関する。

【0002】リード・マラー符号は、誤り検出、誤り訂正が可能な符号として符号化の技術において良く知られている。伝送装置のようにリード・マラー符号により信号を符号化してシリアル形式で伝送する場合、伝送速度が低いと受信側では復号回路を論理回路により構成するよりも回路規模を小さくすることができるROMを使用して誤り訂正を含む復号を行っていた。しかし、伝送速度が高速化すると、それに適応する高速ROMは高価であり、コスト・パフォーマンスの面で使用することができなかった。

【0003】

【従来の技術】リード・マラー符号は、符号化の技術分野において一般に知られている。具体的な参考文献を挙げると、例えば、宮川洋、岩垂好裕、今井秀樹共著、コンピュータ基礎講座18、「符号理論」昭晃堂 (pp.168-176)がある。

40 【0004】最初に、リード・マラー符号の符号化について説明する。リード・マラー符号は線型符号の一種で、一次の符号は陪直交符号であり、この発明に係するのは二元一次の符号であるため、二元一次の符号について説明し、多元の場合は省略する。

【0005】リード・マラー符号の定義の前提として、以下のベクトル積を定義する。

$$a = (a_1, a_2, \dots, a_n), \quad b = (b_1, b_2, \dots, b_n)$$

この二つのベクトルがあった時、ベクトル積を次のように定義する。

50 【0006】

3

4

 $c = a \, b = (a_1 \, b_1, a_2 \, b_2, \dots, a_n \, b_n)$

[0007]

一次のリード・マラー符号の基底は以下のように定義される。但し、 $n = 2^s$ とする。

[数1]

$$v_0 = (v_0^0, v_0^1, \dots, v_0^{n-1})$$

$$v_0^i = 1 \quad i = 0, 1, 2, \dots, 2^s - 1$$

$$v_1 = (v_1^0, v_1^1, \dots, v_1^{n-1})$$

$$v_1^i = 0 \quad 0 \leq i < 2^{s-1}$$

$$v_1^i = 1 \quad \text{その他の } i$$

$$v_k = (v_k^0, v_k^1, \dots, v_k^{n-1})$$

$$v_k^i = 0 \quad j(2^{s-k+1}) \leq i < j(2^{s-k+1}) + 2^{s-k}$$

$$j = 0, 1, 2, \dots, 2^s - 1$$

$$v_k^i = 1 \quad \text{その他の } i$$

【0008】高次の基底はこれら一次の基底のベクトル積で定義され、 r 次の基底は一次の基底 r 個のベクトル積で定義される。従って、符号長 2^s の r 次のリード・マラー符号は $v_0, v_1, v_2, \dots, v_r$ の r 個までの積を基底とした $1 + {}_s C_1 + {}_s C_2 + \dots + {}_s C_r$ 次元の線型ベクトル空間として定義される。例えば、 $S = 5$ の場合、符号長 (n) は 32 となり、 v_0 及び一次の基底は以下になる。

$$v_0 = (11111111111111111111111111111111)$$

$$v_1 = (00000000000000011111111111111111)$$

$$v_2 = (00000000111111110000000011111111)$$

$$v_3 = (00001111000011110000111100001111)$$

$$v_4 = (00110011001100110011001100110011)$$

$$v_5 = (01010101010101010101010101010101)$$

これを一次の符号として使用する場合は、情報を $(a_0, a_1, a_2, a_3, a_4, a_5)$ とすると符号出力 b は、

【0009】

30

$$b = a_0 \, v_0 + a_1 \, v_1 + a_2 \, v_2 + a_3 \, v_3 + a_4 \, v_4 + a_5 \, v_5 \quad (1)$$

または、次のようになる。

【0010】

$$b = a_0 \, v_0 + (a_1 + a_0) \, v_1 + (a_2 + a_0) \, v_2 + (a_3 + a_0) \, v_3 + (a_4 + a_0) \, v_4 + (a_5 + a_0) \, v_5 \quad (2)$$

情報を $(a_0, a_1, a_2, a_3, a_4, a_5) = (0, 0, 1, 0, 0, 1)$ とすると、 $b = v_1 + v_5 = (01010101010101010101010101010101)$ となる。但し、二元なので、加算は排他的論理和になる。

【0011】次にリード・マラー符号の復号について説明する。リード・マラー符号の復号は、最高次の項から順に多数決判定法を用いて行う。上記の 32 ビットの例では式が長大となり説明を簡単にするため、8 ビット ($S = 3$) を例として示し、説明の都合上最高次の項まで含む符号から復号する。

【0012】8 ビットの場合の基底を全て示すと、次の v_0, v_1, v_2, v_3 が一次、 v_{12}, v_{13}, v_{23} が 2 次、 v_{123} が 3 次 (最高次) の式である。

$$v_0 = (11111111)$$

$$v_1 = (00001111)$$

$$v_2 = (00110011)$$

$$v_3 = (01010101)$$

$$v_{12} = v_1 \cdot v_2 = (00000011)$$

$$v_{13} = v_1 \cdot v_3 = (00000101)$$

$$v_{23} = v_2 \cdot v_3 = (00010001)$$

$$v_{123} = v_1 \cdot v_2 \cdot v_3 = (00000001)$$

符号出力 b は、次のようになる。

$$b = a_0 \, v_0 + a_1 \, v_1 + a_2 \, v_2 + a_3 \, v_3 + a_{12} \, v_{12} + a_{13} \, v_{13} + a_{23} \, v_{23} + a_{123} \, v_{123}$$

ここで、 v_0 と他の基底との内積をとると、次のようになる。但し、1 の加算により 1 の数が偶数の場合は 0、奇数の場合は 1 になる。

$$v_0 \cdot v_0 = 0, v_1 \cdot v_0 = 0, v_2 \cdot v_0 = 0, v_3 \cdot v_0 = 0$$

$$v_{12} \cdot v_0 = 0, v_{13} \cdot v_0 = 0, v_{23} \cdot v_0 = 0$$

$$v_{123} \cdot v_0 = 0 + 0 + \dots + 0 + 1 = 1$$

$$v_{123} \cdot v_0 = 0 + 0 + \dots + 0 + 1 = 1$$

50 これは、符号長が変わっても同じ最高次の基底と v_0 と

の内積のみ1でその他は0になることを表す。そこで、最高次の項の復号は、受信符号 $x = (x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7)$ として x と v_0 の内積をとればよい。

$$[0015] \quad x \cdot v_0 = a_{123} \quad v_{123} \cdot v_0 = a_{123} = x_0 + x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7$$

最高次の項は誤り検出も訂正もできないのでこの項の復号はこれで終わる。次に2次の項の復号を行うが、最高次の項の復号ができたので、 x の項から最高次の項を取り除いた $x' = x - a_{123} \cdot v_{123}$ をつくる。この x' と v_1 の内積をとり、 $v_1 \cdot v_1 = v_{11} \cdot v_0$ の関係を用いることにより、 a_{23} が求められる。

$$[0016] \quad x' \cdot v_1 = a_{23} \quad v_{23} \cdot v_1 = a_{23} \quad v_{123} \cdot v_0 = a_{23} = x'_4 + x'_5 + x'_6 + x'_7$$

$$\text{ここで、} v_{23} (v_1 + v_0) = v_{23} \cdot v_1 + v_{23} \cdot v_0 = v_{123} \cdot v_0 \text{ であるから、} x' (v_1 + v_0) = a_{23} v_{123} \cdot v_0$$

$$v_1 \cdot (v_2 + v_0) v_3 = v_1 \cdot v_{23} + v_1 \cdot v_{31} = v_{123} \cdot v_0 \quad (1)$$

$$v_1 \cdot v_2 (v_3 + v_0) = v_1 \cdot v_{23} + v_1 \cdot v_{32} = v_{123} \cdot v_0 \quad (2)$$

$$v_1 \cdot (v_2 + v_0) (v_3 + v_0) = v_1 \cdot v_{23} + v_1 \cdot v_{32} + v_1 \cdot v_{31} + v_1 \cdot v_0 = v_{123} \cdot v_0 \quad (3)$$

$$= v_{123} \cdot v_0 \quad (4)$$

[0020] この性質から、 a_1 を求める場合、次の(5)～(8)に示す4つの式で多数決判定することにより、 a_1 について1ビットの誤り訂正復号ができる。

$$x'' \cdot v_{23} = a_1 \quad v_{123} \cdot v_0 = a_1 = x''_3 + x''_7 \quad (5)$$

$$x'' \cdot (v_2 + v_0) v_3 = a_1 \quad v_{123} \cdot v_0 = a_1 = x''_1 + x''_5 \quad (6)$$

$$x'' \cdot v_2 (v_3 + v_0) = a_1 \quad v_{123} \cdot v_0 = a_1 = x''_2 + x''_6 \quad (7)$$

$$x'' (v_2 + v_0) (v_3 + v_0) = a_1 \quad v_{123} \cdot v_0 = a_1 = x''_0 + x''_4 \quad (8)$$

[0022] 同様に a_2 、 a_3 についても求めることができる。なお、32ビットの場合には、16個ずつ5種類の多数決判定を行い、式が16個あるので7個までの誤りを訂正することができる。

[0023] 最後に a_0 が残るが、同様の方法で、 x'' から一次の項を取り除き、次の式を作る。

$$x''' \cdot v_{123} = a_0 \quad v_{123} \cdot v_0 = a_0 = x'''_7 \quad (9)$$

$$x''' \cdot (v_1 + v_0) v_2 v_3 = a_0 \quad v_{123} \cdot v_0 = a_0 = x'''_3 \quad (10)$$

$$x''' \cdot v_1 (v_2 + v_0) v_3 = a_0 \quad v_{123} \cdot v_0 = a_0 = x'''_5 \quad (11)$$

$$x''' \cdot v_1 v_2 (v_3 + v_0) = a_0 \quad v_{123} \cdot v_0 = a_0 = x'''_6 \quad (12)$$

$$x''' \cdot (v_1 + v_0) (v_2 + v_0) v_3 = a_0 \quad v_{123} \cdot v_0 = a_0 = x'''_1 \quad (13)$$

$$x''' \cdot (v_1 + v_0) v_2 (v_3 + v_0) v_3 = a_0 \quad v_{123} \cdot v_0 = a_0 = x'''_2 \quad (14)$$

$$x''' \cdot v_1 (v_2 + v_0) (v_3 + v_0) = a_0 \quad v_{123} \cdot v_0 = a_0 = x'''_4 \quad (15)$$

$$x''' \cdot (v_1 + v_0) (v_2 + v_0) (v_3 + v_0) = a_0 \quad v_{123} \cdot v_0 = a_0 = x'''_0 \quad (16)$$

[0025] この8個の式から多数決判定を行うことにより a_0 が復号される。ここでは式が8個あるので多数

$$v_{123} \cdot v_0 = a_{23} = x'_0 + x'_1 + x'_2 + x'_3 \text{ となる。}$$

【0017】このように、 a_{23} が二つの式で求められ、この二つの式で多数決をとることで1ビットの誤りを検出できる。同様に a_{12} 、 a_{13} を求めることができる。二次の項の復号ができる。この後、一次の項の復号が、二次の復号と同様に行う。1次の復号では、 x から最高次と二次の項を取り除いた次の式を作る。

$$[0018] \quad x'' = x - a_{123} \cdot v_{123} - (a_{12} v_{12} + a_{13} v_{13} + a_{23} v_{23})$$

最初から二次以上の項が符号として使用されない場合は、 x'' を作る必要はなく、受信符号にそのまま処理を施すことができる。その方法は、二次の場合と同様で、以下の性質を利用して、次の4種類の式(1)～(4)で一次の項を求める。

$$[0019]$$

$$[数2]$$

$$[0021]$$

$$[数3]$$

$$x''' = x'' - (a_1 v_1 + a_2 v_2 + a_3 v_3)$$

最後に残った x''' は $a_0 \cdot v_0$ 項だけなので、同様に書くとして、次の(9)～(16)で示す8個の式になる。

$$[0024]$$

$$[数4]$$

決判定すると3ビットまでの誤りを訂正できてしまうが、一次の符号も使用する場合、 x_i を生成する段階で1ビットの誤りしか訂正できないので、それ以上の誤り訂正は不要である。この8個の式から分かるように、 a_i の復号は、 x_i の各要素 (x_i , $i=0,1,\dots,7$) でそのまま多数決をとっている。その他の符号長の場合も同様である。

【0026】上記のようなリード・マラー符号を伝送装置で使用する場合、図15に示す従来例の構成が採用されている。この従来例では、予め受信符号の 2^s 個のパターンについてソフトウェアシミュレーション等により復号を行い、受信符号と復号結果の対応関係を求め、その対応関係をROMにテーブルとして書き込んでおき、実際の復号を行う時、受信符号からテーブルを引いて復号結果を取り出す技術を用いている。そして、この図15では、受信符号の上位側の符号を並列に展開してラッチ90に、下位側の符号を並列に展開してラッチ91にそれぞれラッチし、ROM92、93はそれぞれ上位、下位の受信符号の各パターン（誤りを含む）をアドレスとして誤り訂正を含む復号結果が格納されている。セレクト94はROM92、93の出力を切り換えて、順番に復号結果を取り出して情報出力を得る。

【0027】

【発明が解決しようとする課題】従来は、リード・マラー符号の復号は規模が小さくなるROMを使用していたが、伝送速度が次第に高速化すると、それに合わせてROMも高速化する必要がある。しかし、高速のROMは高価であり、複数のROMを使用すると復号するための回路のコストが上るといった問題があった。

【0028】本発明はリード・マラー符号の復号を簡単な論理回路を用いてROMを使用せずに高速動作が可能な誤り訂正復号回路を提供することを目的とし、更にリード・マラー符号の半分の符号長で1ビットを表す多数符号を、リード・マラー符号の復号回路を使用して復号可能にすることを別の目的とする。

【0029】

【課題を解決するための手段】図1は本発明の原理構成図である。図1の場合、原信号が $a_0 \sim a_1$ である符号が二元で一次のリード・マラー符号化されて符号長が 2^s ($s=i$)である符号が使用され、その符号を復号するための原理構成を示す。また、符号化信号 x は、上記式(1)の形式に対応する次の式により符号化されているものとする。

$$x = a_0 \cdot v_0 + a_1 \cdot v_1 + \dots + a_{i-1} \cdot v_{i-1} + a_i \cdot v_i$$

図1において、1-1~1-iは多数決方程式計算部、2-1~2-iは n ビット多数決誤り訂正部、3は a_0 の項抽出部、4は a_0 の項についての n ビット多数決誤り訂正を行う n ビット多数決誤り訂正部である。

【0031】リード・マラー符号化された符号入力、

並列に（または直列信号を並列信号に変換して）各多数決方程式計算部1へ入力される。各多数決方程式計算部1では、それぞれ入力する信号（ビット位置）が設定されており、それぞれ、 2^{s-i} 個の排他的論理和を含む論理回路で構成される。各多数決方程式計算部1-1~1-iからの複数（ 2^{s-i} 個）の信号は、それぞれ対応する n ビット多数決誤り訂正部2-1~2-iへ入力する。ここで、各係数 $a_1 \sim a_i$ について、 $n=2^{s-i}-1$ 個の誤りを訂正する。これらの、一次の各係数 $a_1 \sim a_i$ （情報ビット）はそれぞれ出力されると共に a_0 の項抽出部3へ入力される。ここで、 $a_1 \cdot v_1 + a_2 \cdot v_2 + \dots + a_{i-1} \cdot v_{i-1} + a_i \cdot v_i$ が作られ、これを受信した符号入力に加えることにより、 a_0 の項を取り出す。但し、 $v_1 \sim v_i$ はリード・マラー符号の一次の基底である。この a_0 の項抽出部3からの 2^s 個の結果は、 n ビット多数決誤り訂正部4において $n=2^{s-i}-1$ 個の誤りを訂正して、 a_0 の情報ビットが出力される。

【0032】このような構成により、伝送速度が高速化しても、高価なROMを使用することなく、論理回路によりリード・マラー符号の誤り訂正回路を実現することができる。

【0033】なお、リード・マラー符号の符号化が次の式により行われている場合、上記図1により復号した $a_1 \sim a_i$ は $(a_1 + a_0)$ 、 $(a_2 + a_0) \dots (a_{i-1} + a_0)$ 、 $(a_i + a_0)$ であるので、それぞれに a_0 を排他的論理和により加えて、 $a_1 \sim a_i$ を得る処理を行えばよい。

$$x = a_0 \cdot v_0 + (a_1 + a_0) \cdot v_1 + (a_2 + a_0) \cdot v_2 + \dots + (a_{i-1} + a_0) \cdot v_{i-1} + (a_i + a_0) \cdot v_i$$

【0035】

【発明の実施の形態】図2は具体的な構成例、図3は各多数決方程式計算部における計算の内容を示し、図4は各部の構成図、図5は a_0 の項抽出部の構成図である。

【0036】図2において、1~4はそれぞれ上記図1の各符号に対応し、1-1~1-5は多数決方程式計算部、2-1~2-5は7ビット多数決誤り訂正部、3は a_0 の項抽出部、4は a_0 の項の7ビット多数決誤り訂正部である。

【0037】この図2に示す構成は、リード・マラー符号の一次の符号で、 $S=5$ の場合であり、符号長は $2^5=32$ （ビット）で $x_1 \sim x_5$ で表し、情報ビット（復号結果）は6ビットで $a_1 \sim a_6$ で表す。

【0038】各多数決方程式計算部1-1~1-5では、 $x_1 \sim x_5$ の符号入力に対し、 $a_1 \sim a_5$ のそれぞれについて図3に示すような2つのビット入力に対し16個（ $=2^{s-i}$ ）の加算（二元であるため排他的論理和）を行う。図4のA.に情報ビット a_1 に対応する多数決方程式計算部（図2の1-5）の構成を示す。このように16個の排他的論理和回路において、それぞれ決

10

【0049】この動作も、上記図7と同様にセクタ1

4により各多数決方程式計算部10-1~10-iの出力は、時分割的に順番にnビット多数決誤り訂正部11へ供給され、各出力は a_i の項抽出部12へ供給されて a_i の項が抽出されると共に、排他的論理和回路14-1~14-iへ供給されて後処理(a_i を除く)が行われる。

【0050】次に本発明によるリード・マラー符号の復号回路を、情報1ビットで符号長がリード・マラー符号の半分の長さを持つ多数決符号の復号に適用することができる。その原理と構成を以下に説明する。

【0051】リード・マラー符号は符号長を 2^s とすると、 $2^{s-2}-1$ 個の誤りを訂正できる。つまり符号間のハミング距離は 2^{s-1} 個である。この時符号長を 2^{s-1} で $2^{s-2}-1$ 個の誤りを訂正できるのは、1ビットの情報に符号長 2^{s-1} の2個の符号に符号化し、一方の符号が他方の符号の各要素を反転した関係になっている場合ということになる。この符号を、以下、情報1ビットで符号長が 2^{s-1} の多数決符号または 2^{s-1} 多数決符号という。

【0052】例えば、 $S=5$ の場合($2^{s-1}=16$)、1ビットが

(0000111100001111)と(1111000011110000)や(0101101001011010)と(1010010110100101)等のような符号になる。なお、これに対応するリード・マラー符号の符号長は $2^5(=32)$ としているので、符号長は 2^{s-1} になる。

【0053】具体的に、 $S=5$ の場合に、上記の前者の符号を使用して以下のように符号化することができる。

情報=0の時(0000111100001111)に符号化し、

情報=1の時(1111000011110000)に符号化する。

【0054】これを復号する場合、受信符号と情報=0を符号化した符号(ここでは、0000111100001111)の各要素同士の排他的論理和をとって、その結果の各要素で多数決をとればよい。この場合、1が $2^{s-2}+1$ 個以上(この例では9個以上)の場合は1に復号され、1が $2^{s-2}-1$ 個以下(この例では7個以下)であれば0に復号される。1が 2^{s-2} 個(この例では8個)のときは訂正不可となる。

【0055】情報0の(0000111100001111)と情報0の(0000111100001111)の排他的論理和は、(000000000000)で、結果の中に1が0個あるので0に復号される。もし(0000111100001111)に1~7ビットの誤りが入ると、排他的論理和の中に1が1~7個見つかるが、7個以下なので0に復号される。逆に情報1の(111100011110000)と情報0の(0000111100001111)の排他的論理和は、(1111111111111111)で、1が16個あり、1に復号される。同様に(1111000011110000)に1~7ビットの誤りが入ると、排他的論理和の中に1が15~9個みつかるが、9個以上なので1に復号される。

【0056】これは情報iを受信符号を($x_{15}, x_{14}, x_{13},$

$x_{12}, \dots, x_2, x_1, x_0$)として、以下の多数決方程式を計算して7ビット多数決誤り訂正を行う処理に等しい。

$$i=0+x_{15}, i=0+x_{14}, i=0+x_{13}, i=0+x_{12}, i=1+x_{11}$$

$$i=1+x_{10}, i=1+x_9, i=1+x_8, i=0+x_7, i=0+x_6$$

$$i=0+x_5, i=0+x_4, i=1+x_3, i=1+x_2, i=1+x_1$$

$$i=1+x_0$$

10 一方、リード・マラー符号の多数決方程式計算部の計算式が図5に示されている。この中の符号 a_i の計算式を参照すると、受信符号の上位16ビットと下位16ビットの排他的論理和になっている。この符号長が 2^5 の場合でも、同様に受信符号の上位 2^{s-1} ビットと下位 2^{s-1} ビットの排他的論理和になる。

【0057】そこで、受信符号の上位16ビット入力として($x_{31}, x_{30}, \dots, x_{16}, x_{15}, x_{14}, x_{13}, x_{12}, \dots, x_2, x_1, x_0$)=(00001111000011110000)を、下位16ビットとして受信符号($x_{15}, x_{14}, x_{13}, x_{12}, \dots, x_2, x_1, x_0$)を入力すれば、 a_i の出力として多数決復号された結果を出力することができる。この場合、上位と下位は逆でも同様である。また、これを一般化すると、受信符号入力の上位 2^{s-1} ビットか、下位 2^{s-1} ビットに情報=0を符号化した符号を入力し、残りの下位か上位の 2^{s-1} ビットに受信符号を入力すると、 a_i の出力に多数決復号された結果が出力される。

【0058】また、多数決方程式計算部の計算式は受信符号の二つのビットを取り出して排他的論理和をとっているため、ビットの入れ替えを行えば、 a_i 以外の a_j や a_k の復号部でも復号は可能である。

30 【0059】上記の原理を実現する 2^{s-1} 多数決符号とリード・マラー符号を復号する基本構成を図9に示し、図9のタイミングチャートを図10に示す。図9において、30は上位ビットを格納する上位ラッチ、31は下位ビットを格納する下位ラッチ、32は下位セクタ、33はリード・マラー符号復号回路である。リード・マラー符号回路33は、上記図1、図2、図6~図8の何れかの回路で構成することができる。また、下位セクタ32はモード選択信号により切替えられ、符号長 2^5 のリード・マラー符号を復号する時は下位ラッチ31を選択し、 2^{s-1} 多数決符号を復号する時は情報=0の符号化した符号の方を選択する。 2^{s-1} 多数決符号を復号する場合は、リード・マラー符号復号回路33の a_i の出力に多数決復号された結果が出力される。

【0060】図10により図9の動作タイミングを説明すると、受信符号は a_i に示すようにリード・マラー(RM符号と略して表示)(1)の上位、下位、 2^{s-1} 多数決符号、RM符号(2)の上位、下位、…の順に入力される。まず、b.に示すように上位ラッチ30でRM符号(1)の上位がラッチされる。次にc.に示すように下位ラッチ31にRM符号(1)の下位がラッチされる。下

位セクタ 3 2 は e. に示す区間 1 では d. に示すように下位ラッチ 3 1 の方に開いており、区間 1 で RM 符号化 (1) の上位と下位が揃うので、RM 符号 (1) が復号される。

【0061】次に 2^{s-1} 多数決符号が上位ラッチ 3 0 にラッチされる。この時、下位セクタ 3 2 を 2^{s-1} 多数決符号の情報 = 0 の符号の方に開いておくと、区間 2 では 2^{s-1} 多数決符号が復号される。その後、上位ラッチ 3 0 に RM 符号 (2) の上位がラッチされる。次に下位ラッチ 3 1 に RM 符号 (2) の下位がラッチされると、下位セクタ 3 2 が下位ラッチ 3 1 の方に開き、区間 3 では RM 符号 (2) が復号される。なお、 2^{s-1} 多数決符号の復号出力は、リード・マラー符号復号回路 3 3 の一つの符号出力 (例えば a_i) として得られる。

【0062】図 1 1 は 2^{s-1} 多数決符号の復号を含む復号回路の実施例 1 の構成図である。図 1 1 において、多数決方程式計算部 1-1 ~ 1-i, n ビット多数決誤り訂正部 2-1 ~ 2-i, a_0 の項抽出部 3 及び n ビット多数決誤り訂正部 4 の各部分は、上記図 1 と同様であり、入力側の構成は上記図 9 と同様に、受信符号上位ビットの入力と、受信符号下位ビットまたは 2^{s-1} 多数決符号の情報 = 0 の符号が入力されるセクタ 6 (図 9 の下位セクタ 3 2 に対応) を備えている。なお、この構成では、受信符号上位ビット及び受信符号下位ビットは図 9 のようなラッチから出力されてもよい。

【0063】この図 1 1 の動作は、上記図 9、図 1 0 に説明したのと同様にリード・マラー符号の復号 (上記図 6 と同じ) と、 2^{s-1} 多数決符号の復号を行う。なお、 2^{s-1} 多数決符号の復号出力は、n ビット多数決誤り訂正部 2-1 から出力 (a_i の出力) して発生され、図には「他の符号」と表示されている。

【0064】図 1 2 は 2^{s-1} 多数決符号の復号を含む復号回路の実施例 2 の構成図である。図 1 2 の場合、 2^{s-1} 多数決符号の復号を上記図 6 の構成に適用したものであり、図 1 2 の多数決方程式計算部 1 0-1 ~ 1 0-i, n ビット多数決誤り訂正部 1 1-1 ~ 1 1-i, a_0 の項抽出部 1 2 及び n ビット多数決誤り訂正部 1 3, 排他的論理和 1 4-1 ~ 1 4-i の各部分は、上記図 6 と同様であり、入力側に上記図 1 1 と同様の入力⁴⁰が供給されるセクタ 1 5 が設けられている。

【0065】この図 1 2 の構成でも、上記図 6 と同様のリード・マラー符号の復号と、 2^{s-1} 多数決符号の復号を、上記図 9、図 1 0 について説明した方法により行い、 2^{s-1} 多数決符号の復号出力は、上記図 1 1 と同様に n ビット多数決誤り訂正部 2-1 からの出力 (a_i の出力) として発生する。

【0066】図 1 3 は 2^{s-1} 多数決符号の復号を含む復号回路の実施例 3 の構成図である。この構成は、上記図 7 に示す構成により 2^{s-1} 多数決符号の復号を可能とするものであり、図 7 の入力側に上記図 1 1、図 1 2 と同

様に、受信符号上位ビットの入力と、受信符号下位ビットまたは 2^{s-1} 多数決符号の情報 = 0 の符号が入力されるセクタ 6 を備えている。この図 1 3 の構成の動作は、上記図 1 1 と同様でありリード・マラー符号の復号と、 2^{s-1} 多数決符号の復号を行い、 2^{s-1} 多数決符号の復号出力は、共通回路である n ビット多数決誤り訂正部 2 からの a_i の出力として発生する。

【0067】図 1 4 は 2^{s-1} 多数決符号の復号を含む復号回路の実施例 4 の構成図である。この構成は、上記図 8 に示す構成により 2^{s-1} 多数決符号の復号を可能とするものであり、図 8 の入力側に上記図 1 2 と同様に、受信符号上位ビットの入力と、受信符号下位ビットまたは 2^{s-1} 多数決符号の情報 = 0 の符号が入力されるセクタ 1 5 を備えている。この図 1 4 の構成の動作は、上記図 1 1 ~ 図 1 3 と同様でありリード・マラー符号の復号と、 2^{s-1} 多数決符号の復号を行い、 2^{s-1} 多数決符号の復号出力は、共通回路である n ビット多数決誤り訂正部 1 1 からの a_i の出力として発生する。

【0068】上記の説明では入力信号がシリアル²⁰の信号を並列信号に変換した後復号回路へ入力するものとして説明したが、最初から並列な信号として入力しても動作することができる。

【0069】

【発明の効果】本発明によればリード・マラー符号の復号を簡単な論理回路を組み合わせることにより構成することができ、符号信号が高速化しても対応することができる。

【0070】また、n ビット多数決誤り訂正部を時分割で動作させる構成により回路規模を小さくし、回路のコストを低下させることができる。次に 1 ビットを 2^{s-1} の符号長で表す多数決符号を、符号長 2^s のリード・マラー符号の復号回路を用いて復号することを可能とし、重要なビット情報が確実に受け取ることが可能となる。

【図面の簡単な説明】

【図 1】本発明の原理構成図である。

【図 2】具体的な構成例を示す図である。

【図 3】各多数決方程式計算部における計算の内容を示す図である。

【図 4】各部の構成図である。

【図 5】 a_0 の項抽出部の構成図である。

【図 6】各基底に a_0 を含む符号を復号する場合の構成を示す図である。

【図 7】図 1 の構成において n ビット多数決誤り訂正部を共通化した構成を示す図である。

【図 8】図 6 の構成において n ビット多数決誤り訂正部を共通化した構成を示す図である。

【図 9】 2^{s-1} 多数決符号とリード・マラー符号を復号する基本構成を示す図である。

【図 10】図 9 のタイミングチャートを示す図である。

【図 11】 2^{s-1} 多数決符号の復号を含む復号回路の実

施例 1 の構成図である。

【図 1 2】 2^{s-1} 多数決符号の復号を含む復号回路の実施例 2 の構成図である。

【図 1 3】 2^{s-1} 多数決符号の復号を含む復号回路の実施例 3 を構成図である。

【図 1 4】 2^{s-1} 多数決符号の復号を含む復号回路の実施例 4 を構成図である。

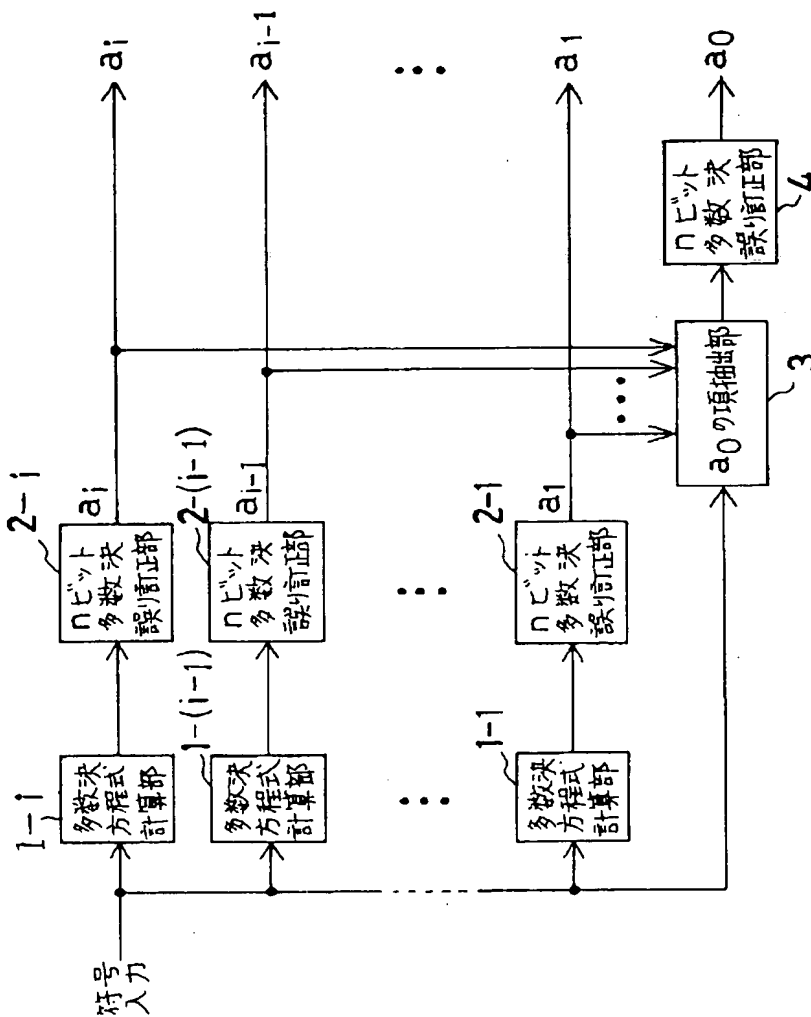
【図 1 5】 従来例の構成図である。

【符号の説明】

- 1-1 ~ 1-i 多数決方程式計算部
 2-1 ~ 2-i n ビット多数決誤り訂正部
 3 a_0 の項抽出部
 4 a_0 の n ビット多数決誤り訂正部

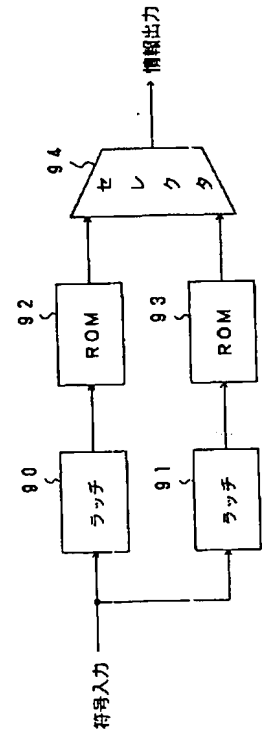
【図 1】

本発明の原理構成図



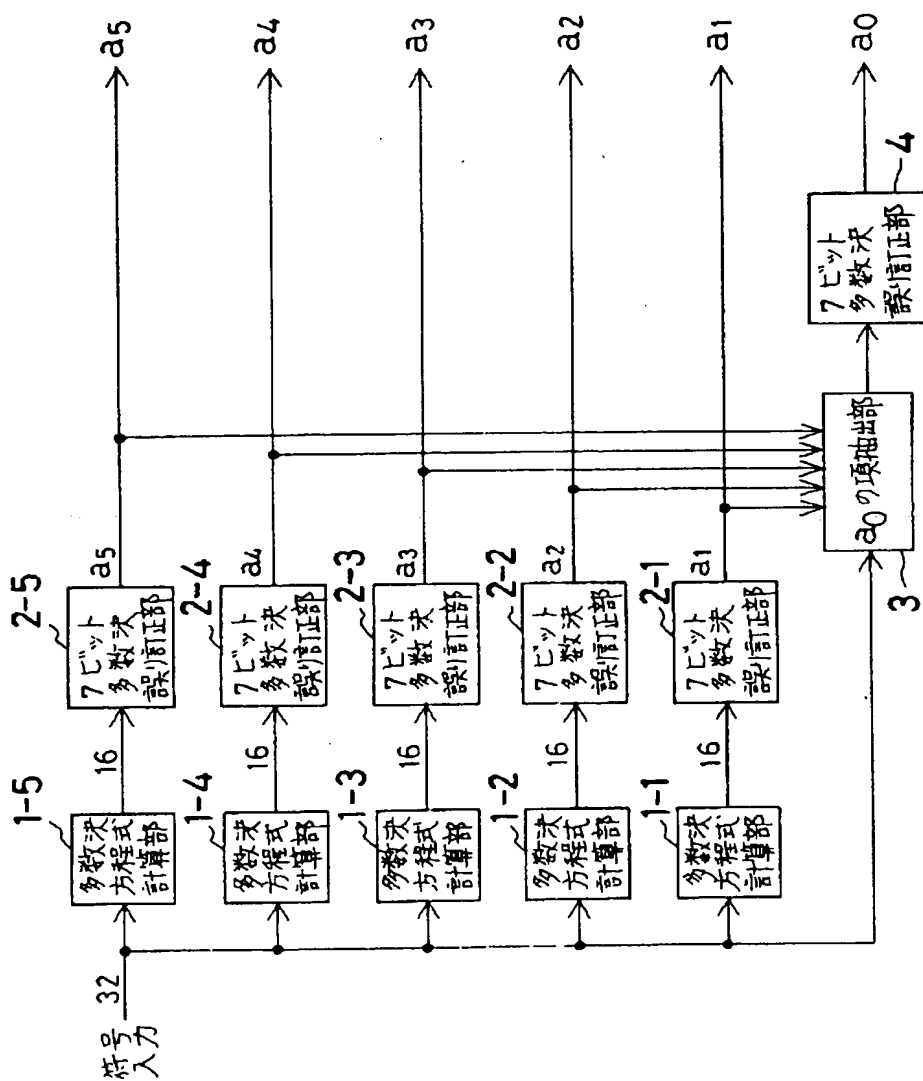
【図 1 5】

従来例の構成図



【図2】

具体的な構成例



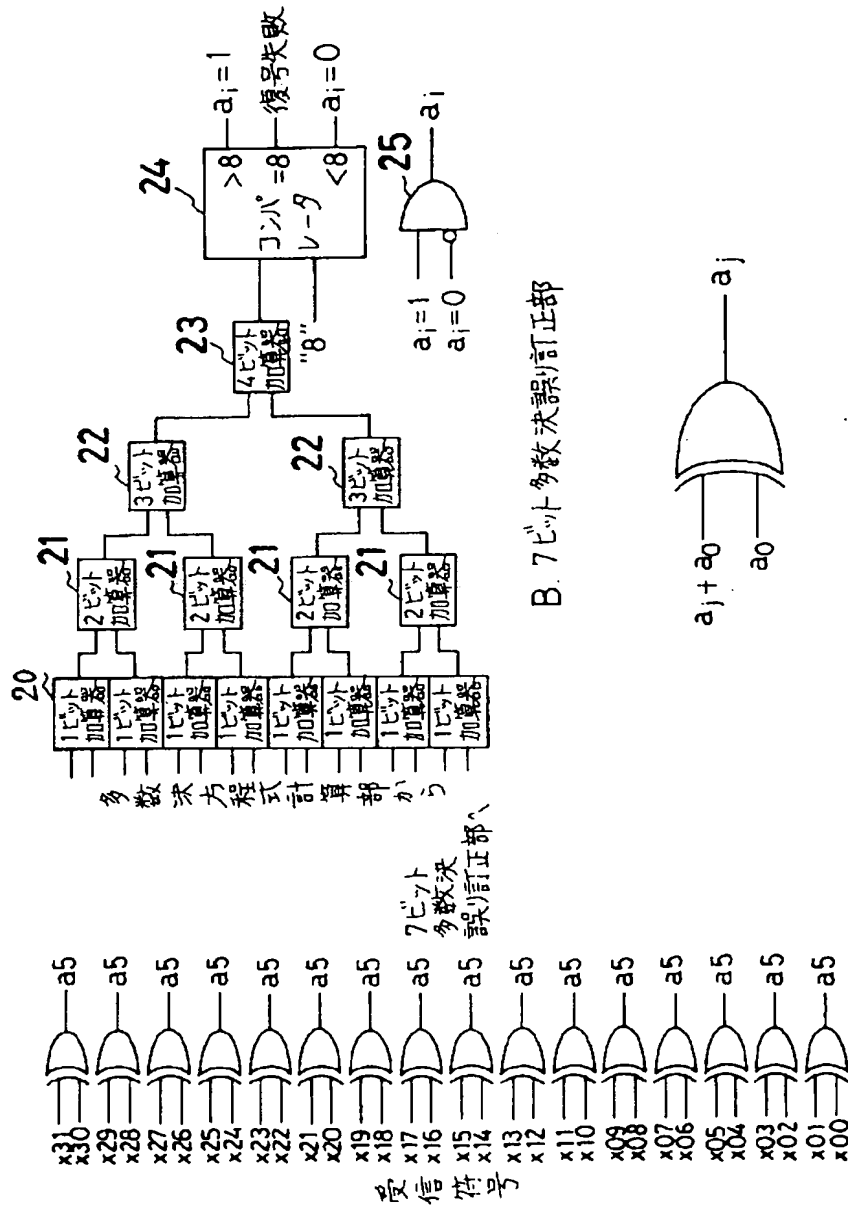
【図3】

各多数決方程式計算部における計算の内容

$$\begin{array}{lllll}
 a_1 = x_{31} + x_{15} & a_2 = x_{31} + x_{23} & a_3 = x_{31} + x_{27} & a_4 = x_{31} + x_{29} & a_5 = x_{31} + x_{30} \\
 a_1 = x_{30} + x_{14} & a_2 = x_{30} + x_{22} & a_3 = x_{30} + x_{26} & a_4 = x_{30} + x_{28} & a_5 = x_{29} + x_{28} \\
 a_1 = x_{29} + x_{13} & a_2 = x_{29} + x_{21} & a_3 = x_{29} + x_{25} & a_4 = x_{27} + x_{25} & a_5 = x_{27} + x_{25} \\
 a_1 = x_{28} + x_{12} & a_2 = x_{28} + x_{20} & a_3 = x_{28} + x_{24} & a_4 = x_{26} + x_{24} & a_5 = x_{25} + x_{24} \\
 a_1 = x_{27} + x_{11} & a_2 = x_{27} + x_{19} & a_3 = x_{23} + x_{19} & a_4 = x_{23} + x_{21} & a_5 = x_{23} + x_{22} \\
 a_1 = x_{26} + x_{10} & a_2 = x_{26} + x_{18} & a_3 = x_{22} + x_{18} & a_4 = x_{22} + x_{20} & a_5 = x_{21} + x_{20} \\
 a_1 = x_{25} + x_9 & a_2 = x_{25} + x_{17} & a_3 = x_{21} + x_{17} & a_4 = x_{19} + x_{17} & a_5 = x_{19} + x_{18} \\
 a_1 = x_{24} + x_8 & a_2 = x_{24} + x_{16} & a_3 = x_{20} + x_{16} & a_4 = x_{18} + x_{16} & a_5 = x_{17} + x_{16} \\
 a_1 = x_{23} + x_7 & a_2 = x_{15} + x_7 & a_3 = x_{15} + x_{11} & a_4 = x_{15} + x_{13} & a_5 = x_{15} + x_{14} \\
 a_1 = x_{22} + x_6 & a_2 = x_{14} + x_6 & a_3 = x_{14} + x_{10} & a_4 = x_{14} + x_{12} & a_5 = x_{13} + x_{12} \\
 a_1 = x_{21} + x_5 & a_2 = x_{13} + x_5 & a_3 = x_{13} + x_9 & a_4 = x_{11} + x_9 & a_5 = x_{11} + x_{10} \\
 a_1 = x_{20} + x_4 & a_2 = x_{12} + x_4 & a_3 = x_{12} + x_8 & a_4 = x_{10} + x_8 & a_5 = x_9 + x_8 \\
 a_1 = x_{19} + x_3 & a_2 = x_{11} + x_3 & a_3 = x_7 + x_3 & a_4 = x_7 + x_5 & a_5 = x_7 + x_6 \\
 a_1 = x_{18} + x_2 & a_2 = x_{10} + x_2 & a_3 = x_6 + x_2 & a_4 = x_6 + x_4 & a_5 = x_5 + x_4 \\
 a_1 = x_{17} + x_1 & a_2 = x_9 + x_1 & a_3 = x_5 + x_1 & a_4 = x_3 + x_1 & a_5 = x_3 + x_2 \\
 a_1 = x_{16} + x_0 & a_2 = x_8 + x_0 & a_3 = x_4 + x_0 & a_4 = x_2 + x_0 & a_5 = x_1 + x_0
 \end{array}$$

【図 4】

各部の構成図

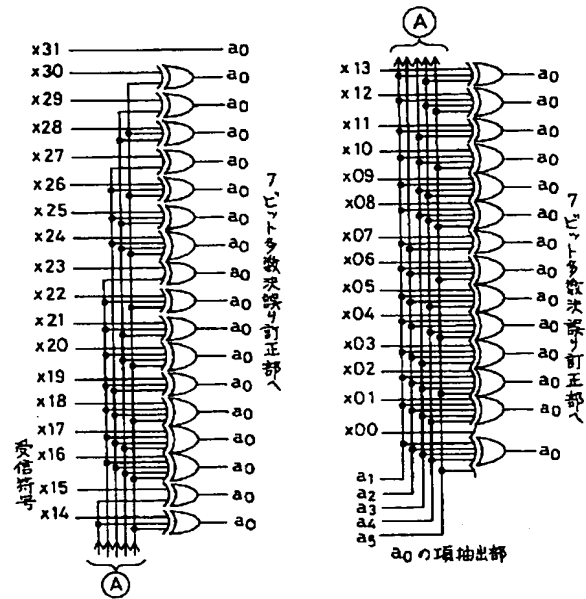


C. 後処理回路

A. 多数決方程式計算部

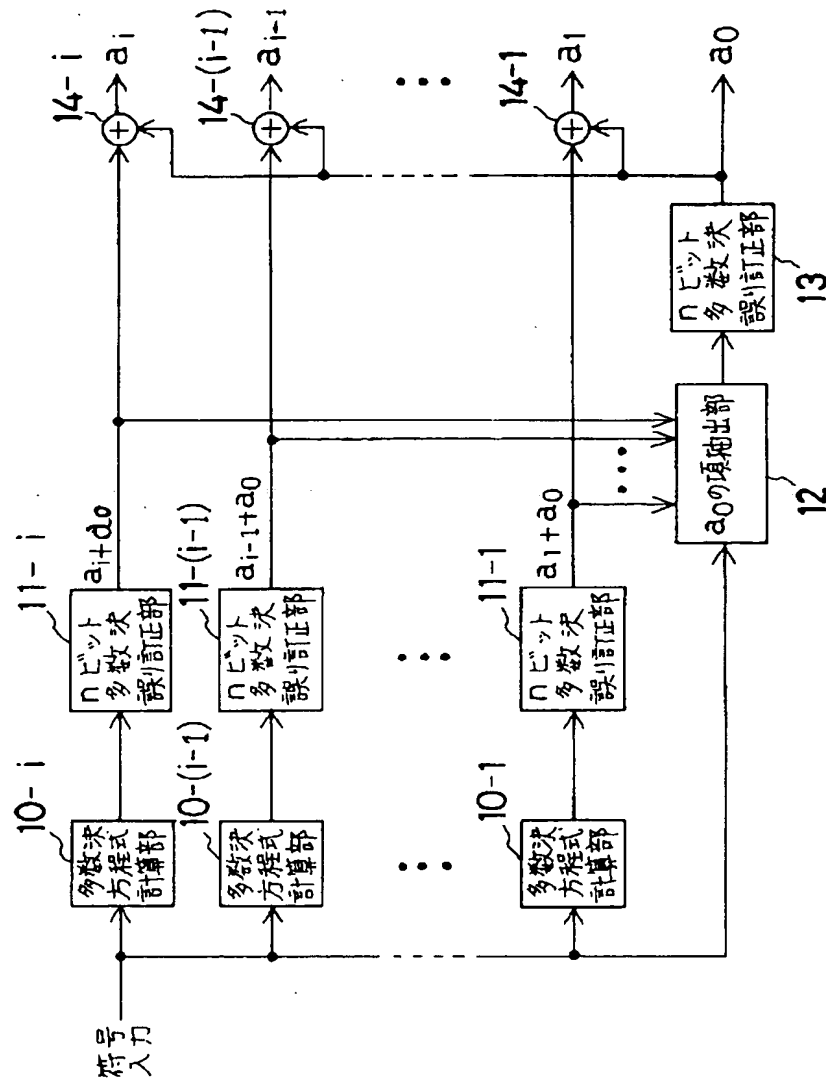
B. 7ビット多数決誤り訂正部

【図 5】

 a_0 の項抽出部の構成図

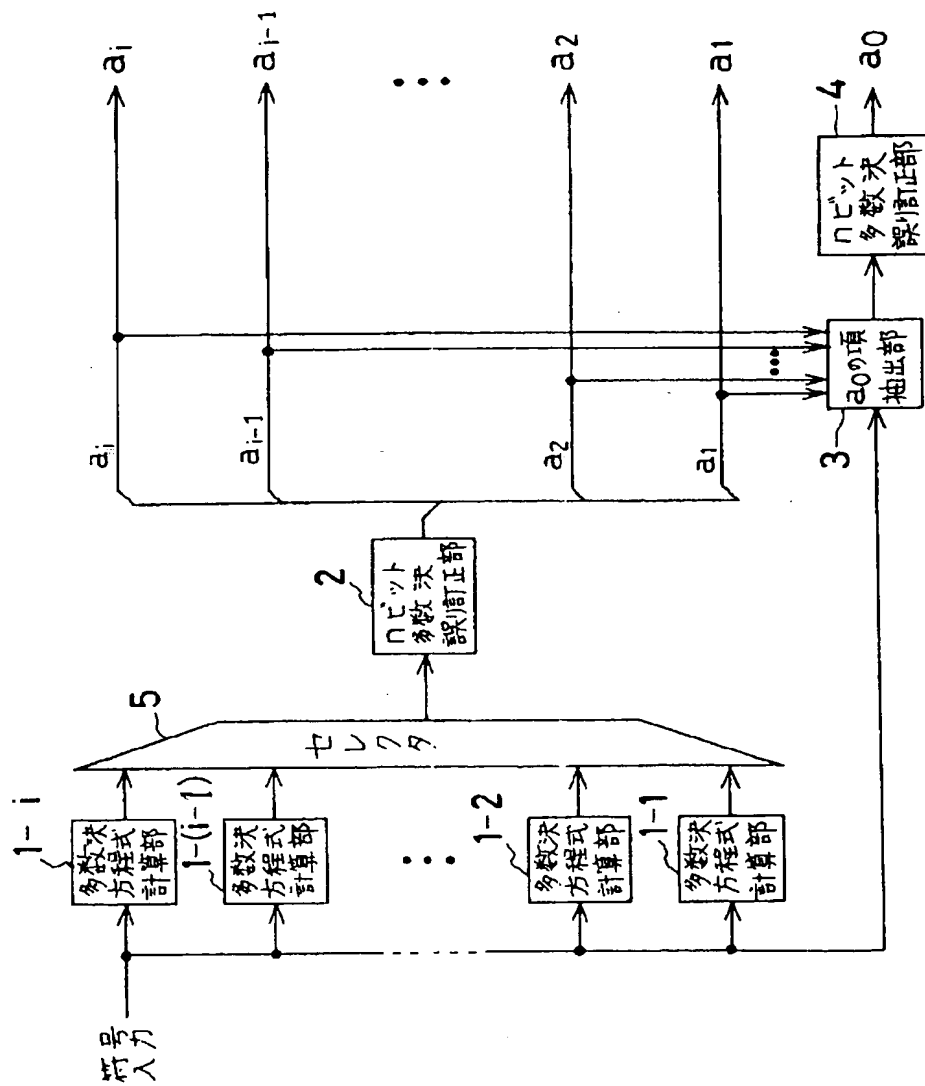
【図6】

各基底に a_0 を含む符号を復号する場合の構成



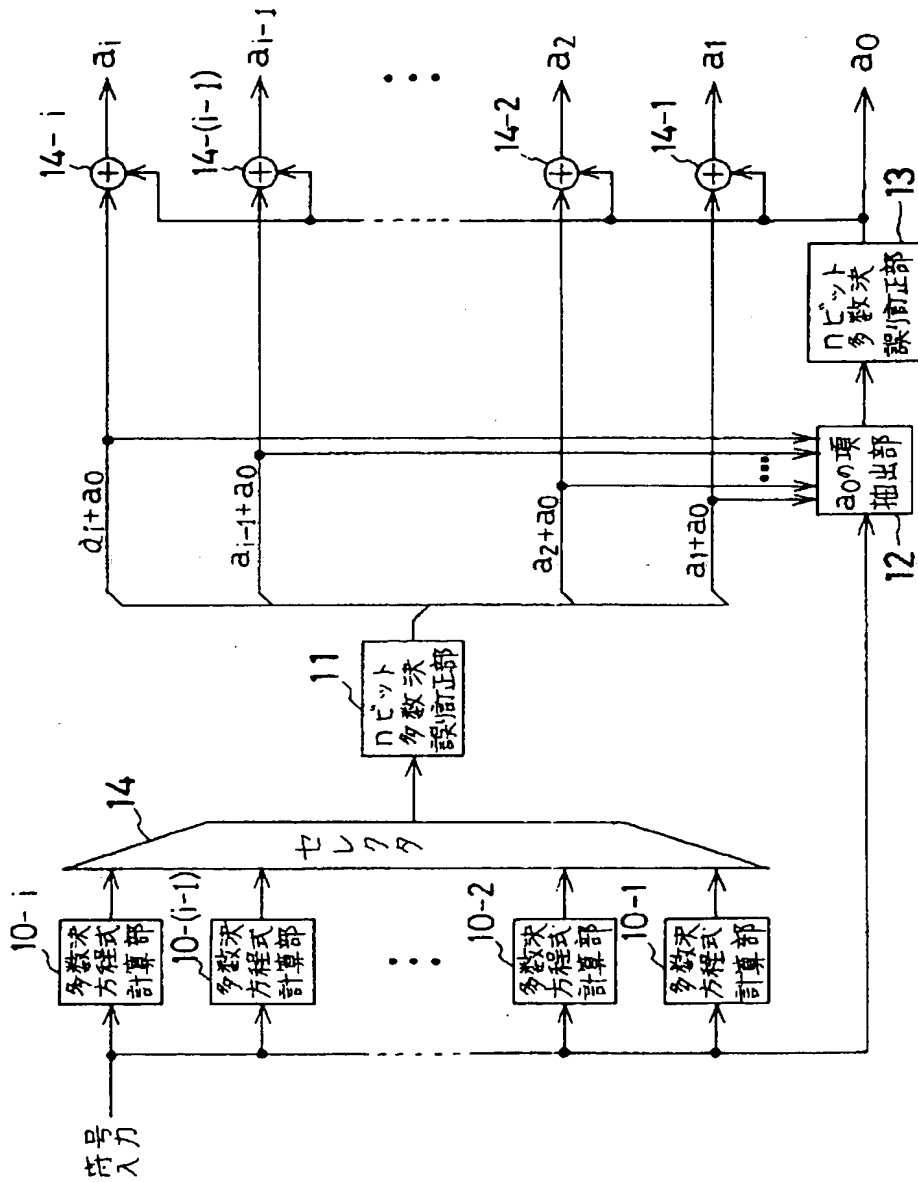
【図7】

図1の構成においてnビット多数決誤り訂正部を
共通化した構成図



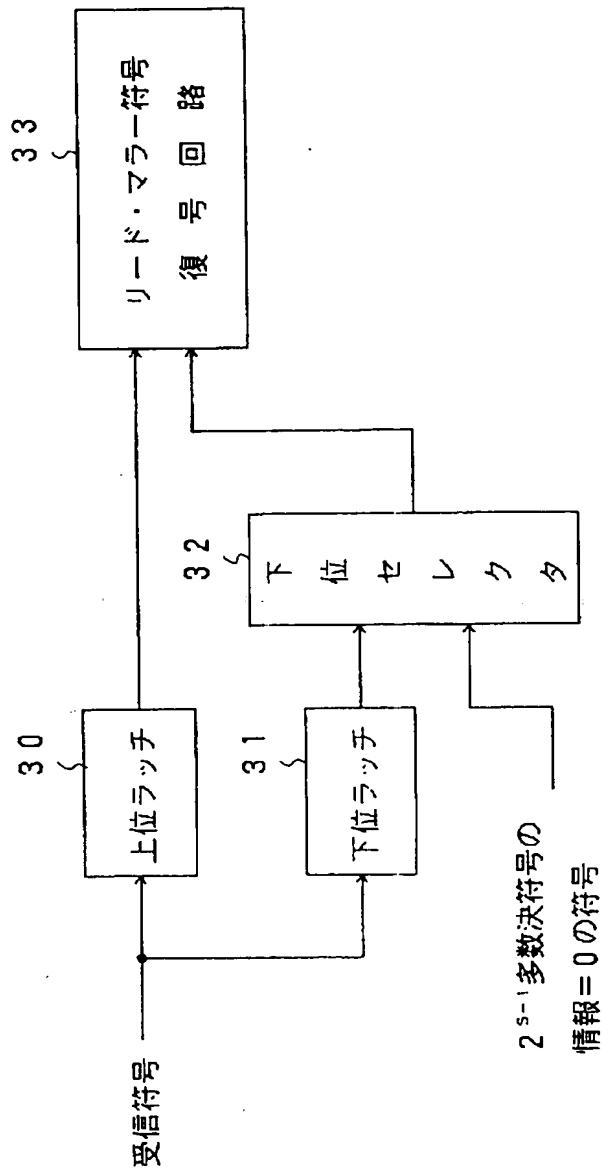
【図 8】

図 6 の構成において n ビット多数決誤り訂正部を
共通化した構成図



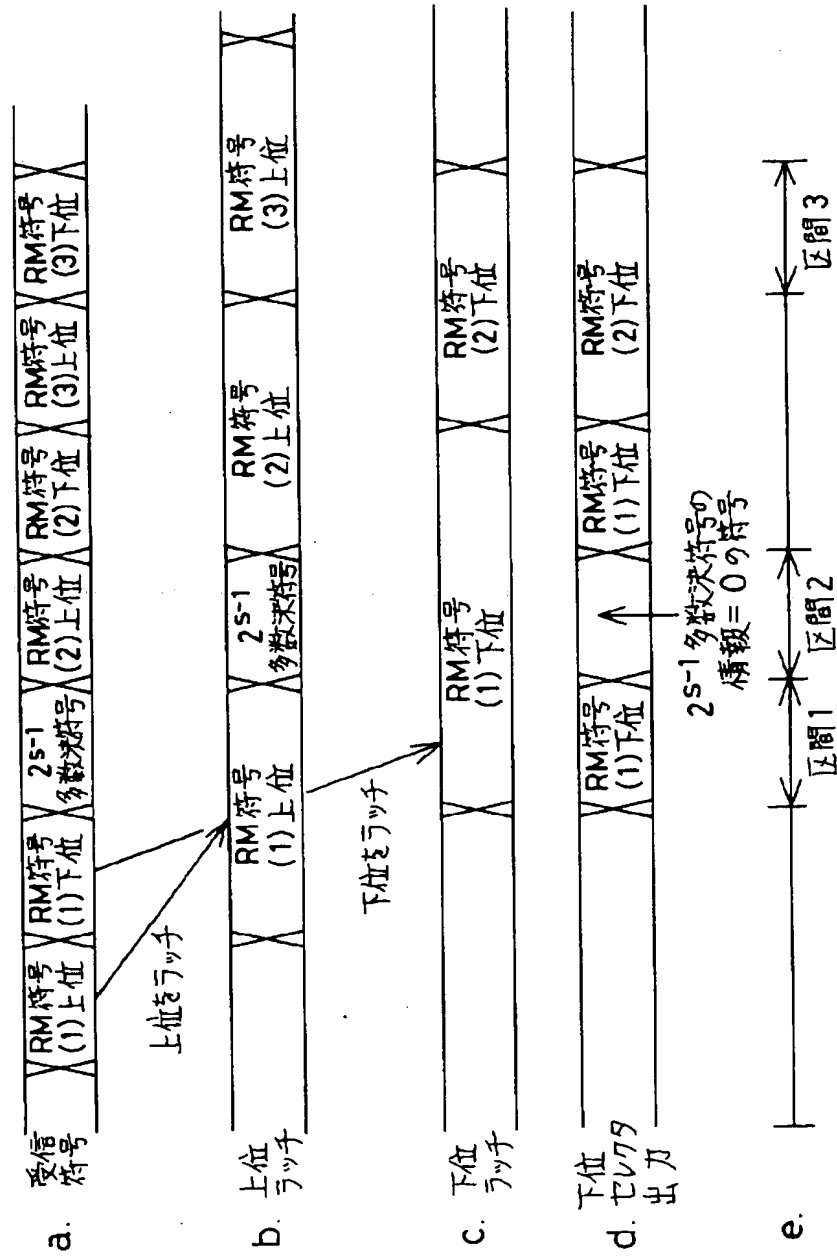
【図 9】

2^{s-1} 多数決符号とリード・マラー符号を復号する基本構成図



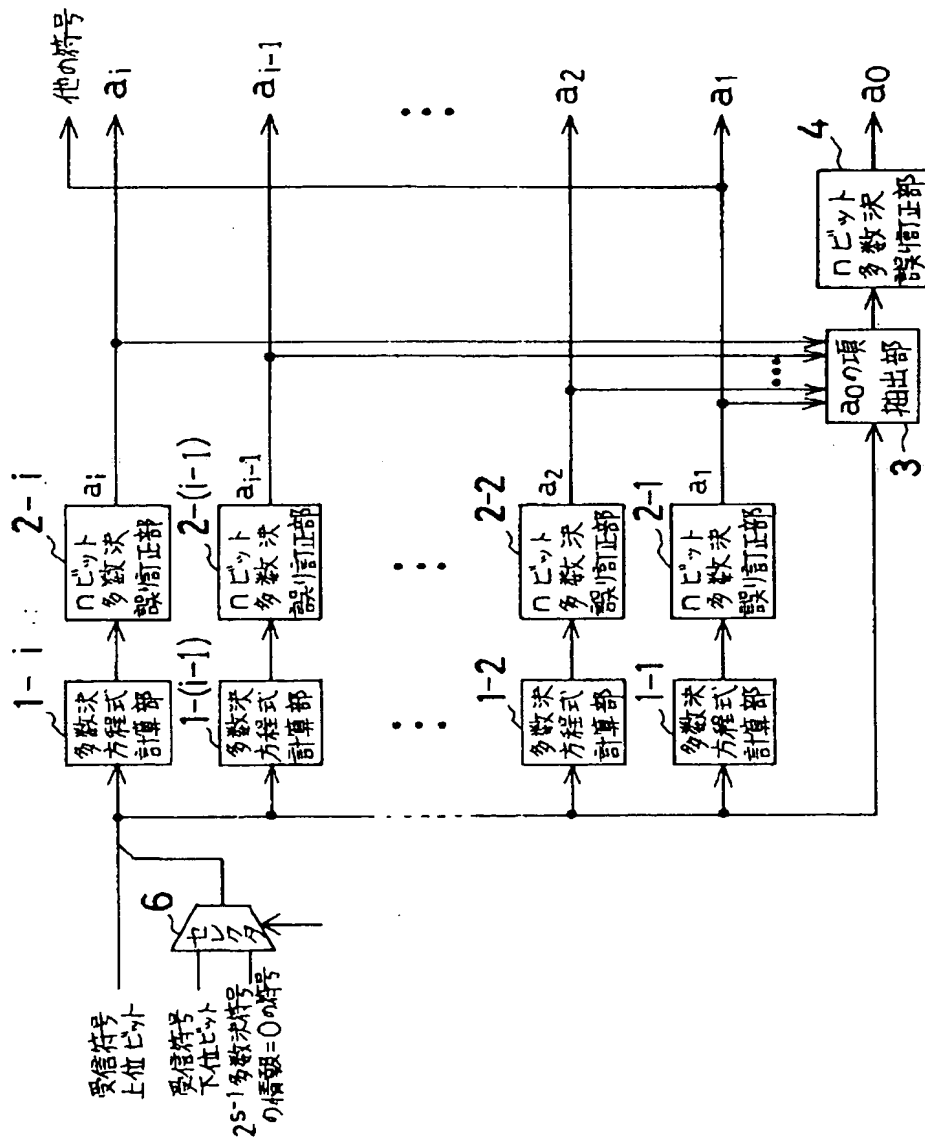
【図 10】

図 9 の タイミング チャート



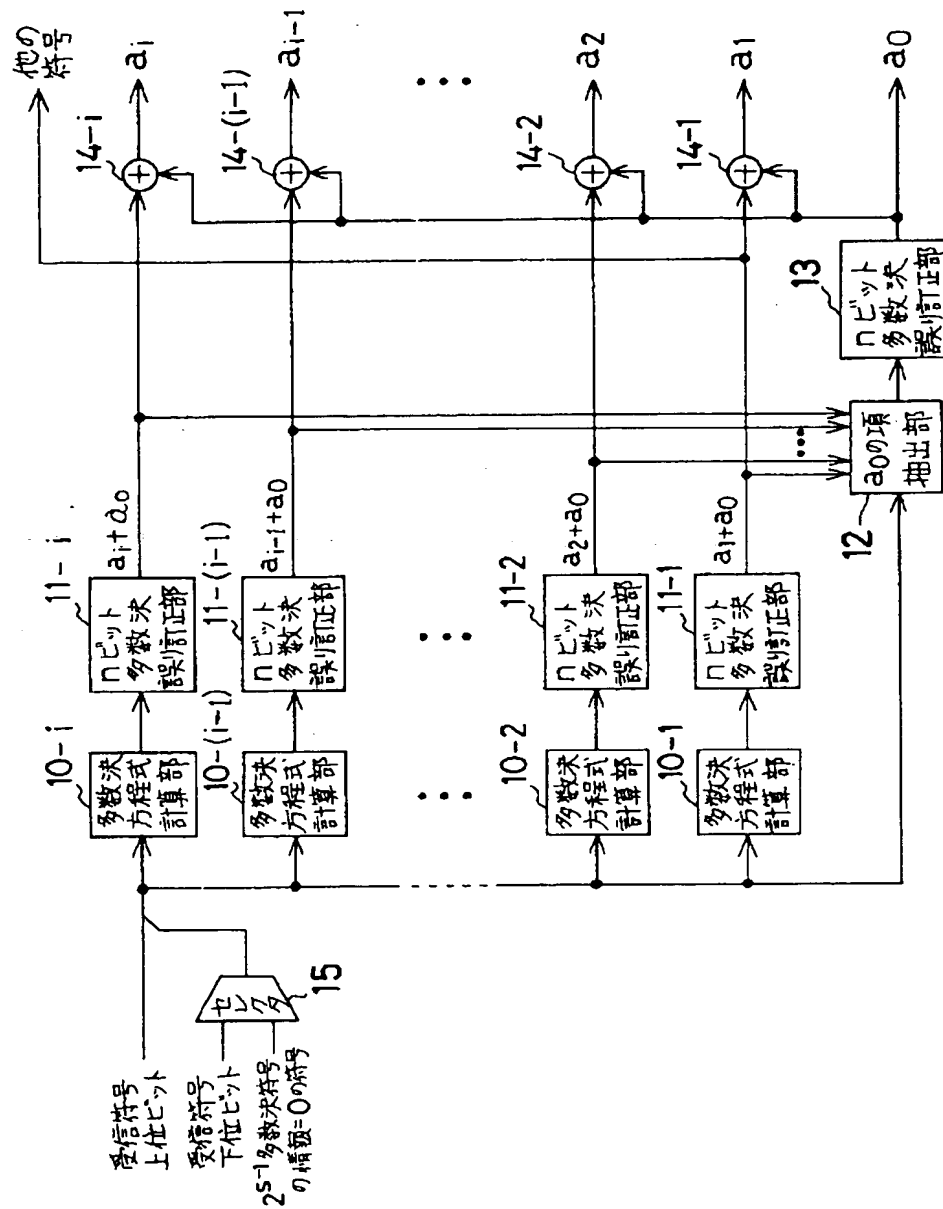
【図 11】

2^{s-1} 多数決符号の復号を含む復号回路の
実施例 1 の構成図



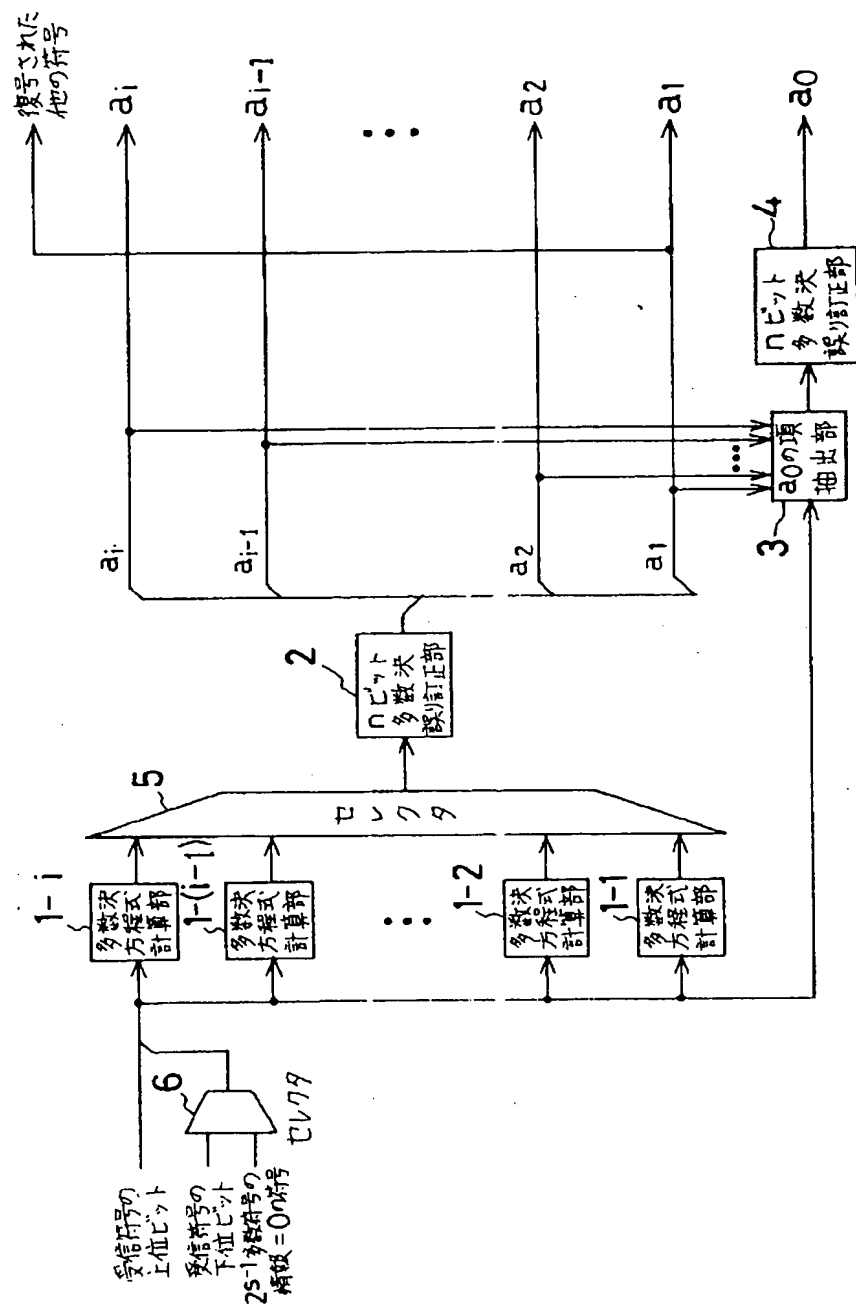
【図 12】

2^{s-1} 多数決符号の復号を含む復号回路の
実施例 2 の構成図



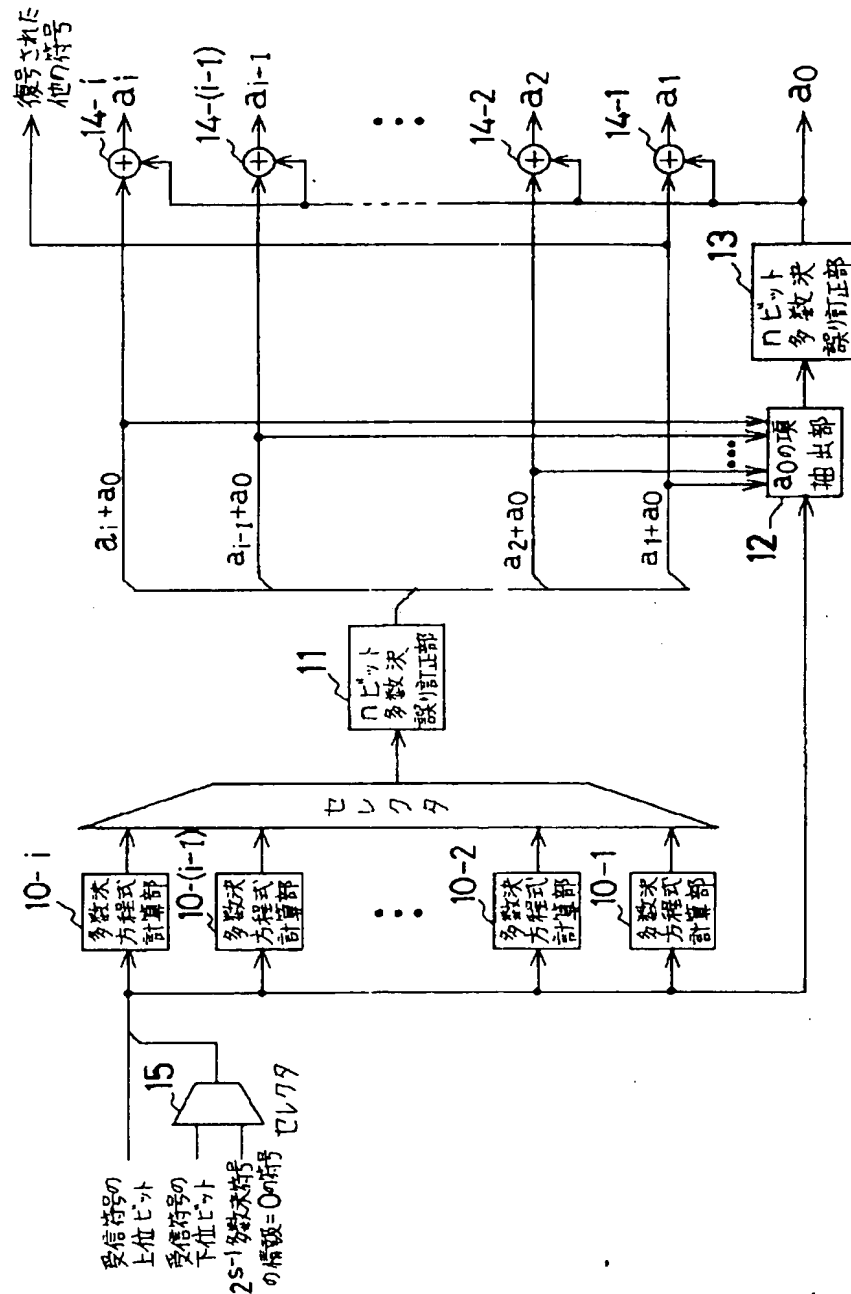
【図 1 3】

2^{s-1} 多数決符号の復号を含む復号回路の
実施例 3 の構成図



【図 1 4】

2^{s-1} 多数決符号の復号を含む復号回路の
実施例 4 の構成図



フロントページの続き

(72)発明者 篠宮 知宏
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(72)発明者 阿比留 節雄
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(72)発明者 ▲廣▼田 正樹
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(72)発明者 宮部 正剛
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内